Amendments to the Claims:

This listing of claims will replace all prior versions and listings of claims in the application.

Listing of Claims:

1. (Currently Amended) A computer system employing a pipeline operation wherein the pipeline is driven by a high clock frequency higher than a low clock frequency by which a critical path instruction in processing data can be executed correctly, comprising:

a high frequency ALU driven by the high clock frequency, a low frequency ALU driven by the low clock frequency, by which, during the low clock frequency, the critical path instruction can be executed correctly, wherein

an execution stage instruction is inputted to both the high frequency ALU and the low frequency ALU,

if the high frequency ALU can execute the execution stage instruction correctly, ~~the~~ an execution result of the high frequency ALU is output as an execution result of a pipeline execution stage, and

if the high frequency ALU can not execute the execution stage instruction correctly, ~~the~~an execution result of the low frequency ALU is output as ~~an~~ the execution result of the pipeline execution stage instead of the execution result of the high frequency ALU.

2. (Currently Amended) The computer system according to claim 1,

wherein the low frequency ALU is composed of ~~plural~~ a plurality of low frequency ALUs;

~~the~~ a low frequency ALU in the plurality of low frequency ALUs in charge of each execution stage is switched in turn, and each of the plurality of low frequency ALUs in charge executes ~~an~~ the execution stage instruction ~~in charge~~ correctly by the low clock frequency which is equal to or lower than ~~the~~ a clock frequency for operating ~~a~~ the critical path instruction correctly.

3. (Currently Amended) The computer system according to claim 2,

wherein ~~the number of~~ the ~~plural~~ plurality of low frequency ALUs is equal to "n" low

frequency ALUs when the ~~pipeline~~ high clock frequency is "n" times of the low clock frequency by which the critical path instruction can be executed correctly,

each of the ~~"n" pieces of the~~ plurality of low frequency ALUs is in charge of "n" pieces of the pipline execution stage ~~execution stages of pipeline~~ in order respectively.

4. (Currently Amended) The computer system according to claim 3, further comprising a comparator comparing the ~~output~~ result of the high frequency ALU and the ~~output~~ result of the low frequency ALU in charge of ~~the~~ a same execution stage for the same instruction,

wherein, the ~~output~~ result of the high frequency ALU is assumed as the execution result of the pipeline execution stage, and when ~~the compared~~ a comparison result of the comparator indicates matching, the ~~output~~ result of the high frequency ALU is determined as the execution result of the execution stage and ~~the~~ pipeline operation is continued, and when the ~~compared~~ comparison result of the comparator indicates mismatching, the ~~output~~ result of the high frequency ALU is replaced with the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage.

5. (Currently Amended) The computer system according to claim 4, wherein when the ~~compared~~ comparison result of the comparator indicates mismatching, all stages of the pipeline are stopped until finishing ~~the~~ a replacement process in which the ~~output~~ result of the low frequency ALU is selected as the execution result of the pipeline execution stage.

6. (Currently Amended) The computer system according to claim 2, further comprising a comparator comparing the ~~output~~ result of the high frequency ALU and the ~~output~~ result of the low frequency ALU in charge of ~~the~~ a same execution stage for ~~the~~ a same instruction,

wherein, the ~~output~~ result of the high frequency ALU is assumed as the execution result of the pipeline execution stage, and when ~~the compared~~ a comparison result of the comparator indicates matching, the ~~output~~ result of the high frequency ALU is determined as the execution result of the execution stage and ~~the~~ pipeline operation is continued, and when the ~~compared~~ comparison result of the comparator indicates mismatching, the ~~output~~ result of the high frequency ALU is replaced with the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage.

7. (Currently Amended) The computer system according to claim 6, wherein when the ~~compared~~ comparison result of the comparator indicates mismatching, all stages of the pipeline are stopped until finishing ~~the~~ a replacement process in which the ~~output~~ result of the low frequency ALU is selected as the execution result of the pipeline execution stage.

8. (Currently Amended) The computer system according to claim 1 further comprising a comparator comparing the ~~output~~ result of the high frequency ALU and the ~~output~~ result of the low frequency ALU in charge of ~~the~~ a same execution stage for ~~the~~ a same instruction,

wherein, the ~~output~~ result of the high frequency ALU is assumed as the execution result of the pipeline execution stage, and when ~~the compared~~ a comparison result of the comparator indicates matching, the ~~output~~ result of the high frequency ALU is determined as the execution result of the execution stage and ~~the~~ pipeline operation is continued, and when the ~~compared~~ comparison result of the comparator indicates mismatching, the ~~output~~ result of the high frequency ALU is replaced with the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage.

9. (Currently Amended) The computer system according to claim 8, wherein when the ~~compared~~ comparison result of the comparator indicates mismatching, all stages of the pipeline are stopped until finishing ~~the~~ a replacement process in which the ~~output~~ result of the low frequency ALU is selected as the execution result of the pipeline execution stage.

10. (Currently Amended) The computer system according to claim 1, further comprising a counter counting ~~the~~ a number of occurrences of ~~the~~ a mismatching detection signal in a predetermined period, and a circuit varying ~~the~~ a pipeline clock frequency according to the ~~counted~~ number of occurrences.

11. (Currently Amended) The computer system according to claim 1, wherein ~~the following~~ amounts of two processes are compared when ~~the~~ a pipeline clock frequency is increased and ~~the~~ a number of the critical path instructions is increased, ~~the one~~ a first amount being an improved process amount of the high frequency ALU and ~~the other~~ a second amount being a deteriorated

process amount ~~by increasing of the~~ that increases if a replacement process ~~of~~ uses the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage when the high frequency ALU cannot execute the execution stage instruction correctly,

wherein, when the ~~former~~ first amount is larger than the ~~latter~~ second amount by ~~the~~ a predetermined amount, the pipeline clock frequency is increased.

12. (Currently Amended) The computer system according to claim 1, wherein~~, the following~~ amounts of two processes are compared when ~~the~~ a pipeline clock frequency is decreased and ~~the~~ a number of the critical path instructions is decreased, ~~the one~~ a first amount being a deteriorated process amount of the high frequency ALU if ~~the~~ a pipeline clock frequency is lowered, and ~~the other~~ a second amount being an improved process amount ~~by decreasing of the~~ that decreases if a replacement process ~~of~~ uses the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage when the high frequency ALU cannot execute the instruction correctly,

wherein, when the ~~latter~~ first amount is larger than the ~~former~~ second amount by a predetermined amount, the pipeline clock frequency is decreased.

13. (Currently Amended) The computer system according to claim 1 further comprising plural ALUs, a data generation circuit generating test data as a critical path data, an execution time measurement circuit measuring the critical path instruction in each ALU, and a detector detecting ~~the fastest~~ which ALU ~~that can execute~~ executes the critical path instruction in a shortest time,

wherein, ~~the~~ a faster ALU detected by the detector is selected as the high frequency ALU, and ~~the other one~~ a slower ALU or ~~plural~~ plurality of slower ALUs is/are selected as the low frequency ALU/ALUs.

14. (Currently Amended) A method for controlling a pipeline operation in a computer system

wherein ~~the~~ a pipeline is driven by a high clock frequency higher than a low clock frequency by which a critical path instruction ~~in processing data~~ can be executed correctly, comprising:

using a high frequency ALU driven by ~~the~~ a high clock frequency, a low frequency ALU driven by ~~the~~ a low clock frequency, by which the low clock frequency executes the critical path

instruction ~~can be executed~~ correctly, wherein

an execution stage instruction is inputted to both the high frequency ALU and the low frequency ALU,

if the high frequency ALU can execute the execution stage instruction correctly, ~~outputting the~~ an execution result of the high frequency ALU is output as an execution result of a pipeline execution stage,

if the high frequency ALU can not execute the execution stage instruction correctly, ~~outputting~~ the execution result of the low frequency ALU is output as ~~an~~ the execution result of the pipeline execution stage instead of the execution result of the high frequency ALU.

15. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 14,

wherein the low frequency ALU is composed of ~~plural~~ a plurality of low frequency ALUs;

switching ~~the~~ one of the plurality of low frequency ~~ALU~~ ALUs in charge of each execution stage in turn, and assigning each of the plurality of low frequency ~~ALU~~ ALUs in charge for an execution stage instruction ~~in charge~~ to execute ~~it~~ the execution stage instruction correctly ~~by~~ during the low clock frequency which is equal to or lower than ~~the~~ a clock frequency for operating ~~a~~ the critical path instruction correctly.

16. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 15,

wherein ~~the number of~~ the ~~plural~~ plurality of low frequency ALUs is equal to "n" low frequency ALUs when the ~~pipeline~~ high clock frequency is "n" times of the low clock frequency by which the critical path instruction can be executed correctly,

each of ~~the "n" pieces of~~ the plurality of low frequency ALUs is in charge of "n" pieces of ~~execution stages of pipeline~~ the pipeline execution stage in order respectively.

17. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 16, further comprising ~~comparing method for~~ comparing the ~~output~~ result of the high frequency ALU and the ~~output~~ result of the low frequency ALU in charge of ~~the~~ a same

execution stage for ~~the~~ a same instruction,

wherein, the ~~output~~ result of the high frequency ALU is assumed as the execution result of the pipeline execution stage, and when ~~the compared~~ a comparison result indicates matching, the ~~output~~ result of the high frequency ALU is determined as the execution result of the execution stage and ~~the~~ pipeline operation is continued, and when the ~~compared~~ comparison result indicates mismatching, the ~~output~~ result of the high frequency ALU is replaced with the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage.

18. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 17, wherein when the ~~compared~~ comparison result indicates mismatching, all stages of the pipeline are stopped until finishing ~~the~~ a replacement process in which the ~~output~~ result of the low frequency ALU is selected as the execution result of the pipeline execution stage.

19. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 15, further comprising ~~a comparing method for~~ comparing the ~~output~~ result of the high frequency ALU and the ~~output~~ result of the low frequency ALU in charge of ~~the~~ a same execution stage for ~~the~~ a same instruction,

wherein, the ~~output~~ result of the high frequency ALU is assumed as the execution result of the pipeline execution stage, and when ~~the compared~~ a comparison result indicates matching, the ~~output~~ result of the high frequency ALU is determined as the execution result of the execution stage and ~~the~~ pipeline operation is continued, and when the ~~compared~~ comparison result indicates mismatching, the ~~output~~ result of the high frequency ALU is replaced with the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage.

20. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 19, wherein when the ~~compared~~ comparison result indicates mismatching, all stages of the pipeline are stopped until finishing ~~the~~ a replacement process in which the ~~output~~ result of the low frequency ALU is selected as the execution result of the pipeline execution stage.

21. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 14 further comprising ~~a comparing method~~ for comparing the ~~output~~ result of the high frequency ALU and the ~~output~~ result of the low frequency ALU in charge of ~~the~~ a same execution stage for ~~the~~ a same instruction,

wherein, the ~~output~~ result of the high frequency ALU is assumed as the execution result of the pipeline execution stage, and when ~~the compared~~ a comparison result indicates matching, the ~~output~~ result of the high frequency ALU is determined as the execution result of the execution stage and ~~the~~ pipeline operation is continued, and when the ~~compared~~ comparison result indicates mismatching, the ~~output~~ result of the high frequency ALU is replaced with the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage.

22. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 21, wherein when the ~~compared~~ comparison result indicates mismatching, all stages of the pipeline are stopped until finishing ~~the~~ a replacement process in which the ~~output~~ result of the low frequency ALU is selected as the execution result of the pipeline execution stage.

23. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 14, further comprising ~~a counting method for~~ counting ~~the~~ a number of occurrences of ~~the~~ a mismatching detection signal in a predetermined period, and ~~a method for~~ varying ~~the~~ a pipeline clock frequency according to the ~~counted~~ number of occurrences.

24. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 14, wherein ~~the following~~ amounts of two processes are compared when ~~the~~ a pipeline clock frequency is increased and ~~the~~ a number of the critical path instructions is increased, ~~the one~~ a first amount being an improved process amount of the high frequency ALU and ~~the other~~ a second amount being a deteriorated process amount ~~by increasing of the~~ that increases if a replacement process ~~of~~ uses the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage when the high frequency ALU cannot execute the instruction correctly,

wherein, when the ~~latter~~ first amount is larger than the ~~former~~ second amount by a

predetermined amount, the pipeline clock frequency is increased.

25. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 14 wherein, ~~the following~~ amounts of two processes are compared when ~~the~~ a pipeline clock frequency is decreased and ~~the~~ a number of the critical path instructions is decreased, ~~the one~~ a first amount being a deteriorated process amount of the high frequency ALU if the pipeline clock frequency is lowered, and ~~the other~~ a second amount being an improved process amount ~~by decreasing of the~~ that decreases if a replacement process ~~of~~ uses the ~~output~~ result of the low frequency ALU as the execution result of the pipeline execution stage when the high frequency ALU cannot execute the instruction correctly,

wherein, when the ~~latter~~ first amount is larger than the ~~former~~ second amount by a predetermined amount, the pipeline clock frequency is decreased.

26. (Currently Amended) The method for controlling a pipeline operation in a computer system according to claim 14, using plural ALUs, further comprising ~~a method for~~ generating test data as a critical path data, ~~a method for~~ measuring the critical path instruction in each ALU, and ~~a method for~~ detecting ~~the fastest~~ which ALU that ~~can execute~~ executes the critical path instruction in a shortest time,

wherein, ~~the~~ a faster detected ALU is selected as the high frequency ALU, and ~~the other one~~ a slower ALU or ~~plural~~ plurality of slower ALUs is/are selected as the low frequency ALU/ALUs.